Real-time Personalization using Embeddings for Search Ranking at Airbnb

Mihajlo Grbovic
Airbnb, Inc.
San Francisco, California, USA
mihajlo.grbovic@airbnb.com

Haibin Cheng
Airbnb, Inc.
San Francisco, California, USA
haibin.cheng@airbnb.com

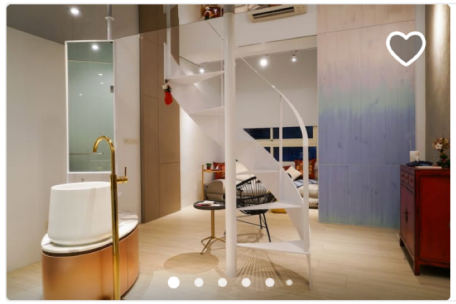
2018KDD best paper

张老师

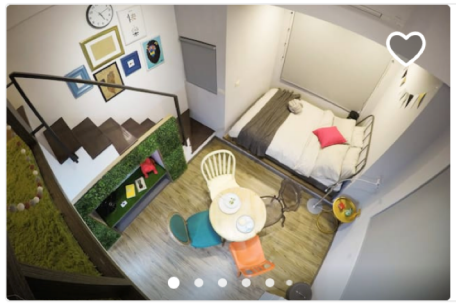
2020.06.21

- Ranking
  - Search Ranking
  - Recommendation Ranking
- Common but specific, unique

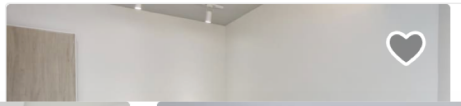
300多处住宿



整间阁楼 · 1室1.5卫2床
Modern Loft 2 min walk to MRT 現代風小套房(捷運中山站步行2分鐘)
4.9分 · 290条评论　超赞房东
"房子设计感很好很喜欢，地理位置方便就在...
¥ 209
每晚
周租优惠



整间阁楼 · 单间1卫1床
Cozy Loft 2 min walk to MRT 溫馨乾淨小套房(捷運中山站步行2分鐘)
4.9分 · 185条评论　超赞房东
"特别好的入住体验，房间设施一应俱全，空...
¥ 174
每晚
周租优惠

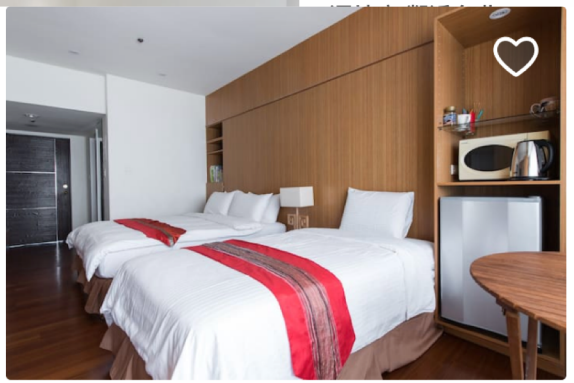


独立房间 · 单间1卫1床
Living in simple life 信義區/市政府捷
¥ 259 ~~¥ 305~~
每晚

您可能还喜欢

独立房间 · 中山区
台北車站捷運站不挑房4人房 4P ROOM
¥ 443 每晚
★★★★☆ 5

独立房间 · 中山区
台北車站捷運站不挑房2人房2P ROOM
¥ 349 每晚
★★★★☆ 19



整套公寓 · 中正区
推薦台北捷運一分鐘柚木裝潢 飯店式公寓
¥ 383 每晚
★★★★½ 327

- Aims for platforms:
  - Clicks and time spent
  - Maximizing conversions
- Optimize search results for both sides of marketplace: supply and demand
  - accommodation (Airbnb)
  - ride sharing (Uber, Lyft)
  - online shops

Content: Airbnb

- Short-term rental market
- Two-sided market: host and guest
  - a user rarely consumes the same item twice
  - a listing(房源) can accept only one guest for a certain set of dates
- Two real-time channels:
  - Search Ranking
  - Similar Listing Recommendations
- Listing and User Embedding techniques

Content: Airbnb

- input query: location and trip dates
- Rank high listings: location, price, style, reviews
- Satisfy: lead days
- Detect listings: would likely reject the guest (bad reviews, pets, length of stay, group size……)-->rank lower

- Short-term interest personalization: list embeddings
- Long-term interest personalization: user type & list type embeddings

Listing embedding

- 1.使用skip-gram构造目标函数 $s = (l_1, \ldots, l_M) \in \mathcal{S}$

$$\mathcal{L} = \sum_{s \in \mathcal{S}} \sum_{l_i \in s} \Big(\sum_{-m \geq j \leq m, i \neq 0} \log \mathbb{P}(l_{i+j} | l_i) \Big).$$

- s为一次会话（session），l_i为一个房源（listing）
- 梯度计算时与l_i的规模有关

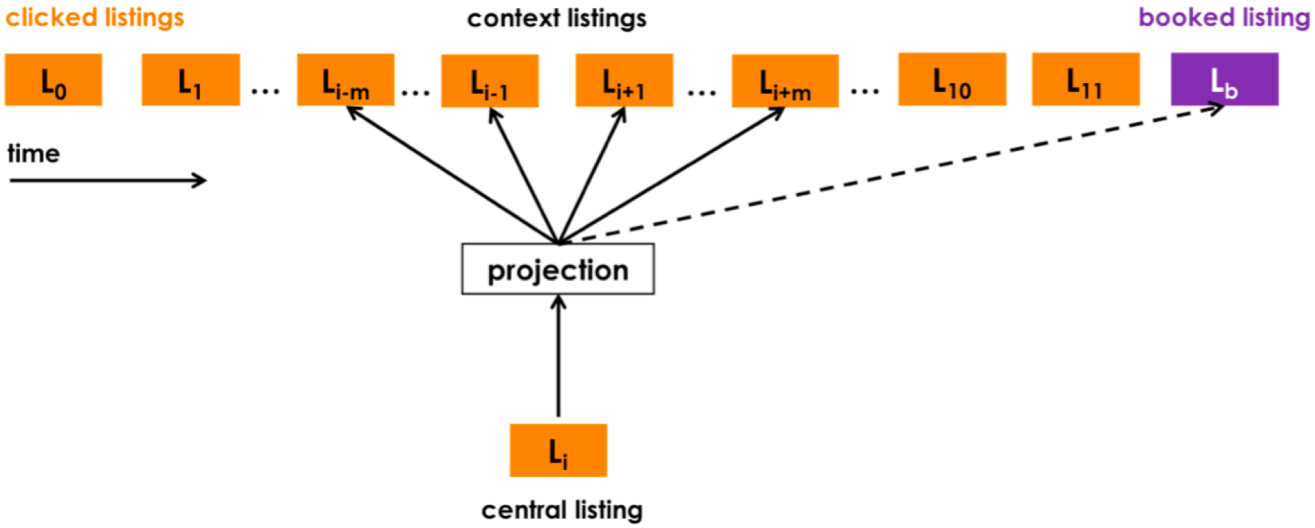
$$P(l_{i+j}|l_i) = \frac{exp(v_{l_i}^T v'_{l_{i+j}})}{\sum_{l=1}^{|v|} exp(v_{l_i}^T v'_l)}$$

Listing embedding

- 2.采用negative sampling构造采样之后的目标函数
- v_l表示位于当前这个listing的embedding向量
- v_c表示当前listing前后context的embedding向量
- (l, c)表示listing pair
- $(l, c) \in D_p$ 正样本
- $(l, c) \in D_n$ 负样本

$$\underset{\theta}{\operatorname{argmax}} \sum_{(l,c)\in\mathcal{D}_p} \log \frac{1}{1+e^{-\mathbf{v}'_c\mathbf{v}_l}} + \sum_{(l,c)\in\mathcal{D}_n} \log \frac{1}{1+e^{\mathbf{v}'_c\mathbf{v}_l}}$$

Listing embedding



- 3. 根据行为构造
有导向于预定的目标函数
- Session:
  - 1)Booked sessions(end with user booking a list)
    - Optimization: predict Neighboring click; eventually book
    - booked listing as global context (无论是否在context window size中)

$$\underset{\theta}{\operatorname{argmax}} \sum_{(l,c)\in\mathcal{D}_p} \log \frac{1}{1+e^{-\mathbf{v}'_c\mathbf{v}_l}} + \sum_{(l,c)\in\mathcal{D}_n} \log \frac{1}{1+e^{\mathbf{v}'_c\mathbf{v}_l}} + \log \frac{1}{1+e^{-\mathbf{v}'_{l_b}\mathbf{v}_l}}$$

v_{l_b}:embedding of the booked listing l_b

  - 2)Exploratory sessions(browsing)：

$$\underset{\theta}{\operatorname{argmax}} \sum_{(l,c)\in\mathcal{D}_p} \log \frac{1}{1+e^{-\mathbf{v}'_c\mathbf{v}_l}} + \sum_{(l,c)\in\mathcal{D}_n} \log \frac{1}{1+e^{\mathbf{v}'_c\mathbf{v}_l}}$$

Listing embedding

- 4.根据房源地区进一步优化得到最终的目标函数
  - 用户通常仅在单个市场（即他们想要停留的位置）内进行搜索
  - Dn很可能主要包含与Dp中的商品不在同一市场的商品
  - 不平衡
- D_{m_n}:当前房源所在市场的房源embedding（负样本）

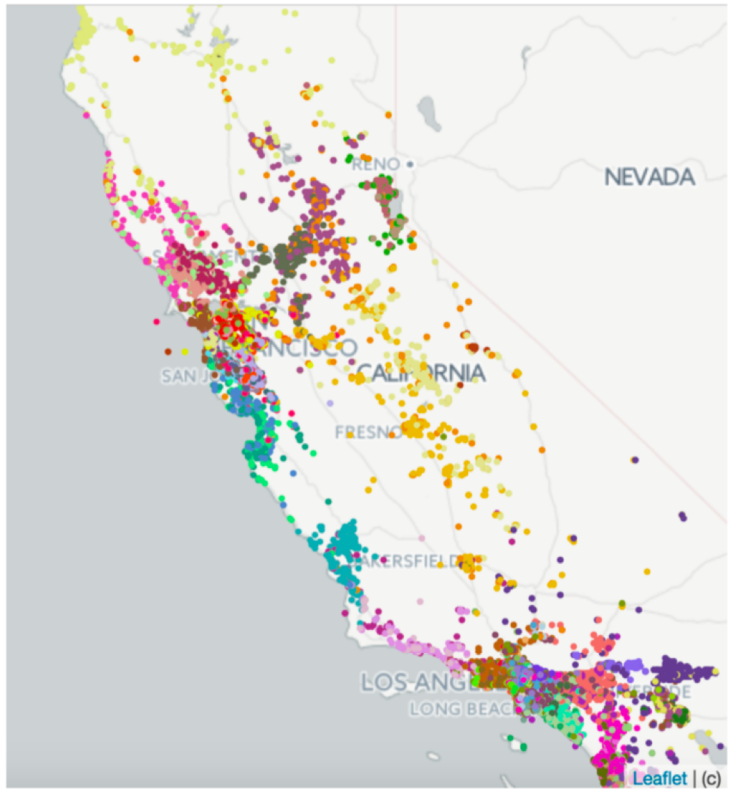
$$\underset{\theta}{\operatorname{argmax}} \sum_{(l,c)\in\mathcal{D}_p} \log \frac{1}{1+e^{-\mathbf{v}'_c \mathbf{v}_l}} + \sum_{(l,c)\in\mathcal{D}_n} \log \frac{1}{1+e^{\mathbf{v}'_c \mathbf{v}_l}}$$

$$+ \log \frac{1}{1+e^{-\mathbf{v}'_{l_b} \mathbf{v}_l}} + \sum_{(l,m_n)\in\mathcal{D}_{m_n}} \log \frac{1}{1+e^{\mathbf{v}'_{m_n} \mathbf{v}_l}}$$

- 解决冷启动：新房源没有会话历史
- 找三个老房源的平均值为 new listing embedding
  - geographically closest
  - of same listing type
  - belong to the same price bucket

构造listing embedding来代表listing是否有效？

- 1 对listing embedding Kmeans聚类



构造listing embedding来代表listing是否有效？

- 2 分类计算平均余弦相似度

- **listing embedding里面**
能够encode到房源的类型
和价格区间这两种features

Table 1: Cosine similarities between different Listing Types

Room Type	Entire Home	Private Room	Shared Room
Entire Home	**0.895**	0.875	0.848
Private Room		**0.901**	0.865
Shared Room			**0.896**

Table 2: Cosine similarities between different Price Ranges

Price Range	<$30	$30-$60	$60-$90	$90-$120	$120+
<$30	**0.916**	0.887	0.882	0.871	0.854
$30-$60		**0.906**	0.889	0.876	0.865
$60-$90			**0.902**	0.883	0.880
$90-$120				**0.898**	0.890
$120+					**0.909**

构造listing embedding来代表listing是否有效？

- 3 k nearest
- **embedding能够提取房源的建筑风格**

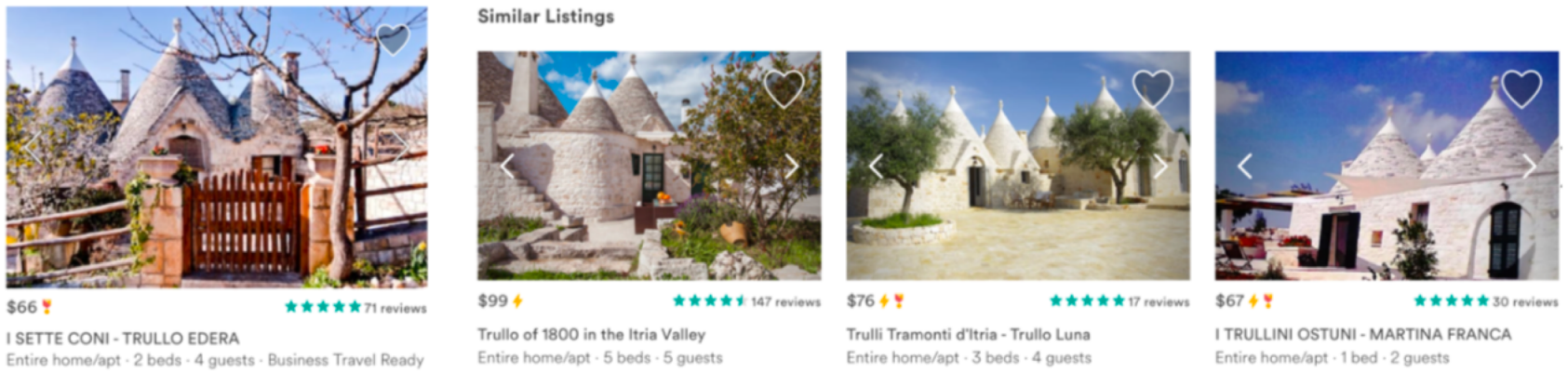

Similar Listings
$66
71 reviews
I SETTE CONI - TRULLO EDERA
Entire home/apt · 2 beds · 4 guests · Business Travel Ready
$99
147 reviews
Trullo of 1800 in the Itria Valley
Entire home/apt · 5 beds · 5 guests
$76
17 reviews
Trulli Tramonti d'Itria - Trullo Luna
Entire home/apt · 3 beds · 4 guests
$67
30 reviews
I TRULLINI OSTUNI - MARTINA FRANCA
Entire home/apt · 1 bed · 2 guests

Figure 3: Similar Listings using Embeddings

搜索阶段如何实现cross market的长期兴趣偏好的挖掘？

- listing embedding不能够解决的问题：
  - 需要基于当前的点击来计算
  - 只提取了用户的短期兴趣
  - 只能用于“相似房源”场景中
  - 只针对同地区下的用户兴趣房源的挖掘

一个user过去是在洛杉矶租过房，但是现在需要在纽约租房

- 解决方案：cross-market similarities：同一用户随时间推移在不同market预定
  - 引入user和listing一起来构建embedding
  - 不局限区域，利用用户所有的历史行为（但最终目标是为了能够让用户下订单，所以从“booking session”来入手）

对于user_type embedding和listing_type embedding的训练使用的数据是book session，而不是listing embedding训练中用到的click session

- 直接将user和listing进行建模存在的问题：
  - booking session远远少于click session
  - 大量的用户历史book量非常少，booking session的长度只有1
  - 大量的listing没怎么被book到，但是skip gram中对物品出现频率是有要求的(5-10)
  - 用户两次预定之间的时间间隔可能非常的长，这期间用户的长期兴趣可能已经改变了
- 解决方式：
  - 针对前三个数据稀疏问题——将listing-id映射为listing-type
  - 针对第四个问题——将用户映射为user-type

从user和listing到user_type和listing_type

- 将user-type和listing-type都映射到同一个特征空间中
  - （user-type， listing-type）
- 实时个性化的体现：同一个listing或user会因为用户行为的发生改变对应的type
- 冷启动问题：新用户可以直接通过特征完成映射

Table 3: Mappings of listing meta data to listing type buckets

Buckets	1	2	3	4	5	6	7	8
Country	US	CA	GB	FR	MX	AU	ES	...
Listing Type	Ent	Priv	Share					
$ per Night	<40	40-55	56-69	70-83	84-100	101-129	130-189	190+
$ per Guest	<21	21-27	28-34	35-42	43-52	53-75	76+	
Num Reviews	0	1	2-5	6-10	11-35	35+		
Listing 5 Star %	0-40	41-60	61-90	90+				
Capacity	1	2	3	4	5	6+		
Num Beds	1	2	3	4+				
Num Bedrooms	0	1	2	3	4+			
Num Bathroom	0	1	2	3+				
New Guest Acc %	<60	61-90	>91					

Table 4: Mappings of user meta data to user type buckets

Buckets	1	2	3	4	5	6	7	8
Market	SF	NYC	LA	HK	PHL	AUS	LV	...
Language	en	es	fr	jp	ru	ko	de	...
Device Type	Mac	Msft	Andr	Ipad	Tablet	Iphone	...	
Full Profile	Yes	No						
Profile Photo	Yes	No						
Num Bookings	0	1	2-7	8+				
$ per Night	<40	40-55	56-69	70-83	84-100	101-129	130-189	190+
$ per Guest	<21	21-27	28-34	35-42	43-52	53-75	76+	
Capacity	<2	2-2.6	2.7-3	3.1-4	4.1-6	6.1+		
Num Reviews	<1	1-3.5	3.6-10	> 10				
Listing 5 Star %	0-40	41-60	61-90	90+				
Guest 5 Star %	0-40	41-60	61-90	90+				

$$\underset{\theta}{\operatorname{argmax}} \sum_{(l,c)\in\mathcal{D}_p} \log \frac{1}{1+e^{-\mathbf{v}'_c\mathbf{v}_l}} + \sum_{(l,c)\in\mathcal{D}_n} \log \frac{1}{1+e^{\mathbf{v}'_c\mathbf{v}_l}}$$

- User type

$$\underset{\theta}{\operatorname{argmax}} \sum_{(u_t,c)\in\mathcal{D}_{book}} \log \frac{1}{1+e^{-\mathbf{v}'_c\mathbf{v}_{u_t}}} + \sum_{(u_t,c)\in\mathcal{D}_{neg}} \log \frac{1}{1+e^{\mathbf{v}'_c\mathbf{v}_{u_t}}},$$

- Listing type

$$\underset{\theta}{\operatorname{argmax}} \sum_{(l_t,c)\in\mathcal{D}_{book}} \log \frac{1}{1+e^{-\mathbf{v}'_c\mathbf{v}_{l_t}}} + \sum_{(l_t,c)\in\mathcal{D}_{neg}} \log \frac{1}{1+e^{\mathbf{v}'_c\mathbf{v}_{l_t}}}$$

- 由于将user_type和listing_type的embedding映射到同一个特征空间中，所以v_c值是一样的

将房东的拒绝行为(rejection)作为负采样的目标函数

存在了用户发出了book request但是房东reject，导致订单没有完成的情况

$$\underset{\theta}{\operatorname{argmax}} \sum_{(u_t,c)\in\mathcal{D}_{book}} \log \frac{1}{1+\exp^{-\mathbf{v}'_c\mathbf{v}_{u_t}}} + \sum_{(u_t,c)\in\mathcal{D}_{neg}} \log \frac{1}{1+\exp^{\mathbf{v}'_c\mathbf{v}_{u_t}}} + \sum_{(u_t,l_t)\in\mathcal{D}_{reject}} \log \frac{1}{1+\exp^{\mathbf{v}'_{l_t}\mathbf{v}_{u_t}}}. \tag{8}$$

$$\underset{\theta}{\operatorname{argmax}} \sum_{(l_t,c)\in\mathcal{D}_{book}} \log \frac{1}{1+\exp^{-\mathbf{v}'_c\mathbf{v}_{l_t}}} + \sum_{(l_t,c)\in\mathcal{D}_{neg}} \log \frac{1}{1+\exp^{\mathbf{v}'_c\mathbf{v}_{l_t}}} + \sum_{(l_t,u_t)\in\mathcal{D}_{reject}} \log \frac{1}{1+\exp^{\mathbf{v}'_{u_t}\mathbf{v}_{l_t}}}. \tag{9}$$

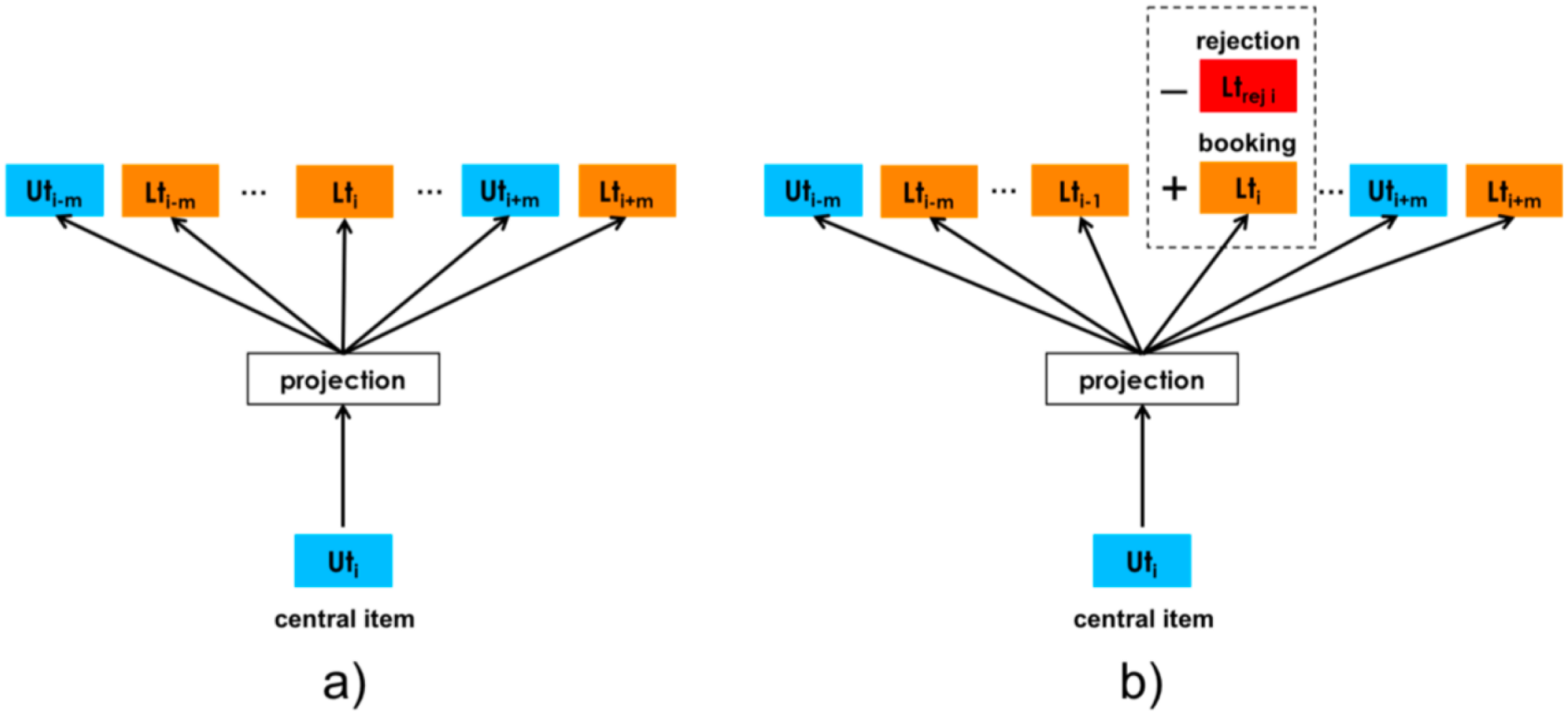


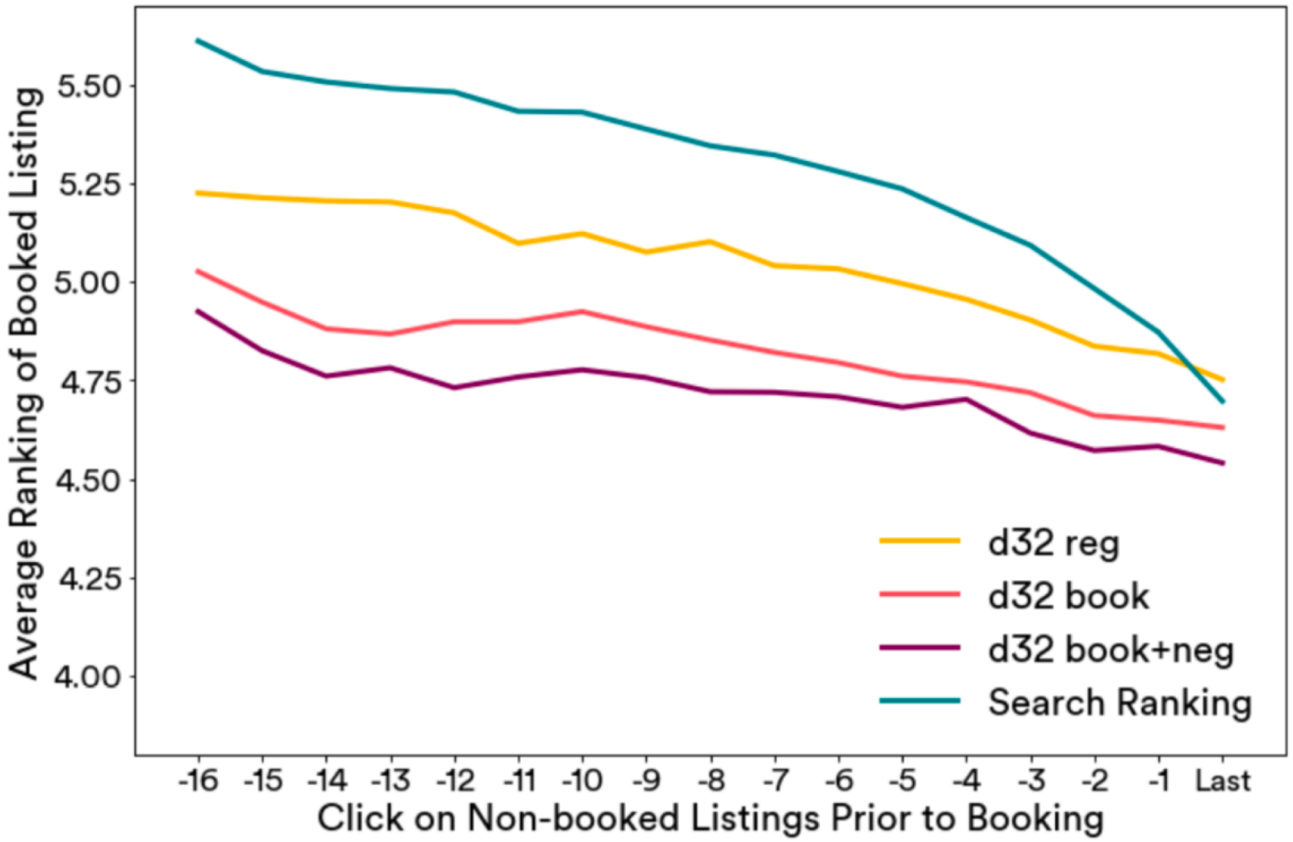
Figure 5: Listing Type and User Type Skip-gram model

根据用户当前用户user_type embedding和候选列表的listing_type embedding之间的余弦相似度，向用户推荐最相关的列表。

Training listing embeddings——Details

- create 800 million click sessions from search, grouping by user id, split based on 30 minute inactivity rule
- remove accidental and short clicks(less than 30 seconds), and kept only sessions consisting of 2 or more clicks
- oversample booked sessions by 5x in the training data
- training data is updated daily in a sliding window manner over multiple months

Offline Evaluation of Listing Embeddings



- search ranking是原有的排序模型。
- re-ranking
  - d32 regular：采用负采样进行训练的方式得到的embedding结果
  - d32 booking global：在d32的基础上加上了book listing作为一个全局的上下文
  - d32 booking global + market negative：在上一个模型的基础上又加入了房源的地区来作为负抽样

Online results

- 应用场景：房源的详情页上的 “相似房源”
- 应用方式：在同地区的候选listing中，计算当前点击listing的embedding值余弦相似度的k-nearest
- 效果：k=12，通过A/Btest发现，embedding方式能够提高21%的点击率，预订订单量增加了4.9%

- 原有的排序模型：
  - 算法：pairwise的支持lambda rank的GBDT
  - 特征：listing features、user features、query features和cross features
- 现在的排序模型：
  - 算法：pairwise的支持lambda rank的GBDT
  - 特征：listing features、user features、query features和cross features，及新增embedding features

Search Ranking

- step1：准备三类embedding值（listing embedding，user_type embedding、listing_type embedding）
- step2：获取基础指标（均基于用户历史点击的listing）
- step3：提取地区embedding
- step4：获取embedding features

$$UserTypeListingTypeSim(u_t, l_t) = cos(\mathbf{v}_{u_t}, \mathbf{v}_{l_t}).$$

$$EmbClickSim(l_i, H_c) = \max_{m \in M} cos(\mathbf{v}_{l_i}, \sum_{l_h \in m, l_h \in H_c} \mathbf{v}_{l_h}),$$

- step5：在原有的排序模型中加入embedding features

(1) H_c: **clicked listing_ids** - listings that user clicked on in last 2 weeks.
(2) H_{lc}: **long-clicked listing_ids** - listing that user clicked and stayed on the listing page for longer than 60 sec.
(3) H_s: **skipped listing_ids** - listings that user skipped in favor of a click on a lower positioned listing
(4) H_w: **wishlisted listing_ids** - listings that user added to a wishlist in last 2 weeks.
(5) H_i: **inquired listing_ids** - listings that user contacted in last 2 weeks but did not book.
(6) H_b: **booked listing_ids** - listings that user booked in last 2 weeks.

Feature Name	Description
EmbClickSim	similarity to clicked listings in H_c
EmbSkipSim	similarity to skipped listings H_s
EmbLongClickSim	similarity to long clicked listings H_{lc}
EmbWishlistSim	similarity to wishlisted listings H_w
EmbInqSim	similarity to contacted listings H_i
EmbBookSim	similarity to booked listing H_b
EmbLastLongClickSim	similarity to last long clicked listing
UserTypeListingTypeSim	user type and listing type similarity

Embedding feature在airbnb搜索系统上的测评

Feature Name	Coverage	Feature Importance
EmbClickSim	76.16%	**5/104**
EmbSkipSim	78.64%	**8/104**
EmbLongClickSim	51.05%	20/104
EmbWishlistSim	36.50%	47/104
EmbInqSim	20.61%	12/104
EmbBookSim	8.06%	46/104
EmbLastLongClickSim	48.28%	11/104
UserTypeListingTypeSim	86.11%	22/104

总结

- 希望embedding表达什么？
  - Session定义：短期兴趣（listing embedding）和长期兴趣（user-type embedding和listing-type embedding）
  - 根据业务需求……
- 如何让embedding向量学到东西？
  - Word2vec的context构建（global），多目标融合策略……
- 数据稀疏怎么办？
  - 频次过滤，聚类……
- 如何评估向量的效果？
  - 离线：聚类、可视化；排序看效果
  - 线上：实时个性化
    - 多实体embedding向量空间一致性
    - Negative反馈